US009338285B2

(12) **United States Patent**
Anderson et al.

(10) **Patent No.: US 9,338,285 B2**
(45) **Date of Patent: May 10, 2016**

(54) **METHODS AND SYSTEMS FOR MULTI-FACTOR AUTHENTICATION IN SECURE MEDIA-BASED CONFERENCING**

(71) Applicant: **Edifire LLC**, Boston, MA (US)

(72) Inventors: **Eric Anderson**, Grafton, MA (US); **Daniel P. Goepp**, Boston, MA (US)

(73) Assignee: **Edifire LLC**, Boston, MA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **14/834,142**

(22) Filed: **Aug. 24, 2015**

(65) **Prior Publication Data**

US 2015/0365522 A1 Dec. 17, 2015

Related U.S. Application Data

(63) Continuation of application No. 14/480,091, filed on Sep. 8, 2014, now Pat. No. 9,118,809, which is a continuation-in-part of application No. 14/289,122, filed on May 28, 2014, now Pat. No. 8,970,660.

(60) Provisional application No. 61/889,926, filed on Oct. 11, 2013.

(51) **Int. Cl.**

H04N 7/15	(2006.01)
H04M 3/38	(2006.01)
H04M 3/56	(2006.01)
H04L 29/06	(2006.01)
H04L 12/18	(2006.01)

(52) **U.S. Cl.**

CPC ***H04M 3/38*** (2013.01); ***H04L 12/18*** (2013.01); ***H04L 12/1818*** (2013.01); ***H04L 65/1069*** (2013.01); ***H04L 65/1076*** (2013.01); ***H04L 65/1093*** (2013.01); ***H04L 65/4038*** (2013.01); ***H04L 65/607*** (2013.01); ***H04M 3/56*** (2013.01); ***H04M 3/567*** (2013.01); ***H04N 7/152*** (2013.01); ***H04N 7/155*** (2013.01); *H04M 2203/6009* (2013.01); *H04M 2203/6054* (2013.01)

(58) **Field of Classification Search**

CPC H04N 7/14; H04N 7/141; H04N 7/142; H04N 7/15; H04N 7/152

USPC 348/14.01, 14.08, 14.09; 370/260–261

See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

6,262,978	B1	7/2001	Bruno et al.
6,404,873	B1	6/2002	Beyda et al.
6,529,941	B2	3/2003	Haley et al.
6,532,079	B1	3/2003	Serex et al.

(Continued)

OTHER PUBLICATIONS

(C1) International Search Report and Written Opinion from corresponding International PCT patent application No. PCT/US15/49011, dated Oct. 15, 2015, 10 pages.

(C2) Fuze Meeting Video Conferencing—Boardroom quality HD video conferencing to any internet connected device, Fuze Meeting, San Francisco, CA, www.fuzemeeting.com, 5 pages.

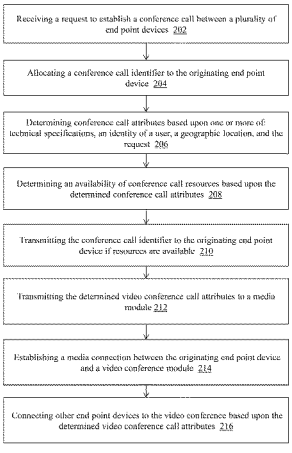
Primary Examiner — Stella L Woo

(74) *Attorney, Agent, or Firm* — Proskauer Rose LLP

(57) **ABSTRACT**

Methods and apparatuses, including computer program products, are described for multi-factor authentication of media-based (e.g., video and/or audio) conferencing between a plurality of end point devices. The methods and apparatuses provide for analysis of an end point media stream using a matrix of authentication factors, where the authentication factors include user-specific factors, environment factors, and technical factors, to determine an authentication score for the first end point device.

26 Claims, 8 Drawing Sheets



(56) **References Cited**

U.S. PATENT DOCUMENTS

6,633,324 B2 10/2003 Stephens, Jr.
6,850,985 B1 * 2/2005 Giloi et al. 709/230
7,031,700 B1 4/2006 Weaver et al.
7,058,168 B1 6/2006 Knappe et al.
7,113,992 B1 9/2006 Even
7,165,060 B2 1/2007 Foster et al.
7,174,365 B1 2/2007 Even et al.
7,243,123 B1 7/2007 Allen et al.
7,312,809 B2 12/2007 Bain et al.
7,353,251 B1 4/2008 Balakrishnan
7,428,000 B2 9/2008 Cutler et al.
7,456,858 B2 11/2008 Schrader et al.
7,466,812 B1 12/2008 Mahy et al.
7,532,231 B2 5/2009 Pepperell et al.
7,539,155 B1 5/2009 Bajwa et al.
7,590,230 B1 9/2009 Surazski
7,653,705 B2 1/2010 Gudipaty et al.
7,656,824 B2 2/2010 Wang et al.
7,657,021 B2 2/2010 Flockhart et al.
7,792,063 B2 9/2010 Dahle
7,916,322 B2 3/2011 Pineau
7,937,442 B2 5/2011 Sekaran et al.
8,079,079 B2 12/2011 Zhang et al.
8,149,261 B2 4/2012 Sarkar et al.
8,259,152 B2 9/2012 Erikkson et al.
8,300,557 B2 10/2012 Sekaran et al.
8,359,616 B2 1/2013 Rosenberg et al.
8,379,819 B2 2/2013 Diskin et al.
8,380,790 B2 2/2013 Lee et al.
8,390,669 B2 3/2013 Catchpole et al.
8,433,050 B1 4/2013 Baten et al.
8,468,581 B2 6/2013 Cuende Alonso
8,487,976 B2 7/2013 Kenoyer
8,489,887 B1 7/2013 Newman et al.
8,526,336 B2 9/2013 Sarkar et al.
8,588,111 B1 11/2013 Kridlo
8,605,878 B2 12/2013 Claudatos et al.
8,817,668 B2 8/2014 Sekaran et al.
8,887,067 B2 11/2014 Tripathi et al.
8,941,741 B1 1/2015 McDonough et al.
8,943,568 B1 * 1/2015 Hromi et al. 726/7
8,955,048 B2 2/2015 Uchida et al.
9,118,809 B2 * 8/2015 Anderson et al. 1/1
2001/0040532 A1 11/2001 Yasuda
2001/0054071 A1 12/2001 Loeb
2002/0010008 A1 * 1/2002 Bork et al. 455/567
2002/0029350 A1 3/2002 Cooper et al.
2002/0065079 A1 5/2002 Ekman et al.
2002/0071529 A1 6/2002 Nelkenbaum
2002/0136166 A1 9/2002 Armbruster et al.
2002/0136167 A1 9/2002 Steele et al.
2002/0147814 A1 10/2002 Kimchi et al.
2002/0159394 A1 10/2002 Decker et al.
2003/0018717 A1 1/2003 Haley et al.
2003/0033271 A1 2/2003 Hendricks
2003/0055974 A1 3/2003 Brophy et al.
2003/0070072 A1 4/2003 Nassiri
2003/0081114 A1 5/2003 Matthews et al.
2003/0149724 A1 8/2003 Chang
2003/0169330 A1 9/2003 Ben-Shachar et al.
2003/0232614 A1 12/2003 Squibbs
2004/0047342 A1 3/2004 Gavish et al.
2004/0103147 A1 5/2004 Flesher et al.
2004/0107270 A1 6/2004 Stephens et al.
2004/0131184 A1 7/2004 Wu et al.
2004/0153648 A1 8/2004 Rotholtz et al.
2004/0169722 A1 9/2004 Pena
2004/0207724 A1 10/2004 Crouch et al.
2004/0224665 A1 11/2004 Kokubo
2004/0225728 A1 11/2004 Huggins et al.
2005/0018826 A1 1/2005 Benco et al.
2005/0021713 A1 1/2005 Dugan et al.
2005/0055211 A1 3/2005 Claudatos et al.
2005/0108435 A1 5/2005 Nowacki et al.
2005/0120122 A1 6/2005 Farnham
2005/0193137 A1 9/2005 Farnham
2005/0276406 A1 12/2005 Keohane et al.
2006/0004820 A1 1/2006 Claudatos et al.
2006/0146124 A1 7/2006 Pepperell et al.
2006/0164507 A1 7/2006 Eshkoli et al.
2006/0167815 A1 7/2006 Peinado et al.
2006/0259755 A1 * 11/2006 Kenoyer 713/1
2006/0285670 A1 12/2006 Chin et al.
2007/0025537 A1 2/2007 Claudatos et al.
2007/0047715 A1 3/2007 Madhusudan et al.
2007/0071213 A1 3/2007 Claudatos et al.
2007/0094199 A1 4/2007 Deshpande et al.
2007/0124760 A1 5/2007 Novak et al.
2007/0127463 A1 6/2007 Dahle
2007/0172045 A1 7/2007 Nguyen et al.
2007/0201449 A1 8/2007 Restrick, Jr. et al.
2007/0285504 A1 12/2007 Hesse
2008/0008458 A1 1/2008 Gudipaty et al.
2008/0072159 A1 3/2008 Cockerton
2008/0159490 A1 7/2008 Gaudin et al.
2008/0168162 A1 7/2008 Chen et al.
2008/0183467 A1 7/2008 Yuan et al.
2008/0195664 A1 8/2008 Maharajh et al.
2008/0219426 A1 9/2008 Lai
2008/0263010 A1 10/2008 Roychoudhuri et al.
2008/0281914 A1 11/2008 Miyata
2008/0293396 A1 11/2008 Barnes et al.
2009/0028316 A1 1/2009 Jaiswal et al.
2009/0132822 A1 5/2009 Chen
2009/0179983 A1 7/2009 Schindler
2009/0220064 A1 9/2009 Gorti et al.
2009/0220066 A1 9/2009 Shaffer et al.
2009/0232364 A1 9/2009 Hosoi
2009/0300520 A1 12/2009 Ashutosh et al.
2009/0327247 A1 12/2009 Jia et al.
2010/0042647 A1 2/2010 Schultz et al.
2010/0061539 A1 3/2010 Cloran et al.
2010/0124322 A1 5/2010 Bill
2010/0128641 A1 5/2010 Karnalkar et al.
2010/0153497 A1 6/2010 Sylvain et al.
2010/0205667 A1 8/2010 Anderson et al.
2010/0306283 A1 12/2010 Johnson et al.
2011/0030028 A1 2/2011 Menten et al.
2011/0165858 A1 * 7/2011 Gisby et al. 455/411
2011/0268263 A1 11/2011 Jones et al.
2011/0273526 A1 11/2011 Mehin et al.
2011/0279634 A1 11/2011 Periyannan et al.
2011/0287748 A1 11/2011 Angel et al.
2012/0036195 A1 2/2012 Kennedy
2012/0051719 A1 3/2012 Marvit
2012/0053936 A1 3/2012 Marvit
2012/0072499 A1 3/2012 Cipolli et al.
2012/0124138 A1 5/2012 Smith et al.
2012/0140681 A1 6/2012 Kaminsky et al.
2012/0147127 A1 6/2012 Satterlee et al.
2012/0262530 A1 10/2012 Whitfield et al.
2012/0278155 A1 11/2012 Faith
2012/0287228 A1 11/2012 Mishra et al.
2013/0027504 A1 1/2013 Zhang
2013/0106989 A1 5/2013 Gage et al.
2013/0133049 A1 5/2013 Peirce
2013/0163580 A1 6/2013 Vass
2013/0282446 A1 10/2013 Dobell
2013/0321562 A1 12/2013 Takahashi
2013/0329970 A1 * 12/2013 Irie et al. 382/118
2013/0335515 A1 12/2013 Wamorkar et al.
2014/0022889 A1 1/2014 Syrett et al.
2014/0104371 A1 4/2014 Calman et al.
2014/0111596 A1 * 4/2014 Grevers, Jr. 348/14.01
2014/0112211 A1 4/2014 Walters
2014/0150059 A1 5/2014 Uchida et al.
2014/0165152 A1 * 6/2014 Farouki 726/4
2014/0289326 A1 9/2014 McCormack et al.
2014/0289834 A1 * 9/2014 Lindemann 726/7
2014/0317061 A1 10/2014 Rao et al.
2015/0135101 A1 * 5/2015 Ellis et al. 715/762

* cited by examiner

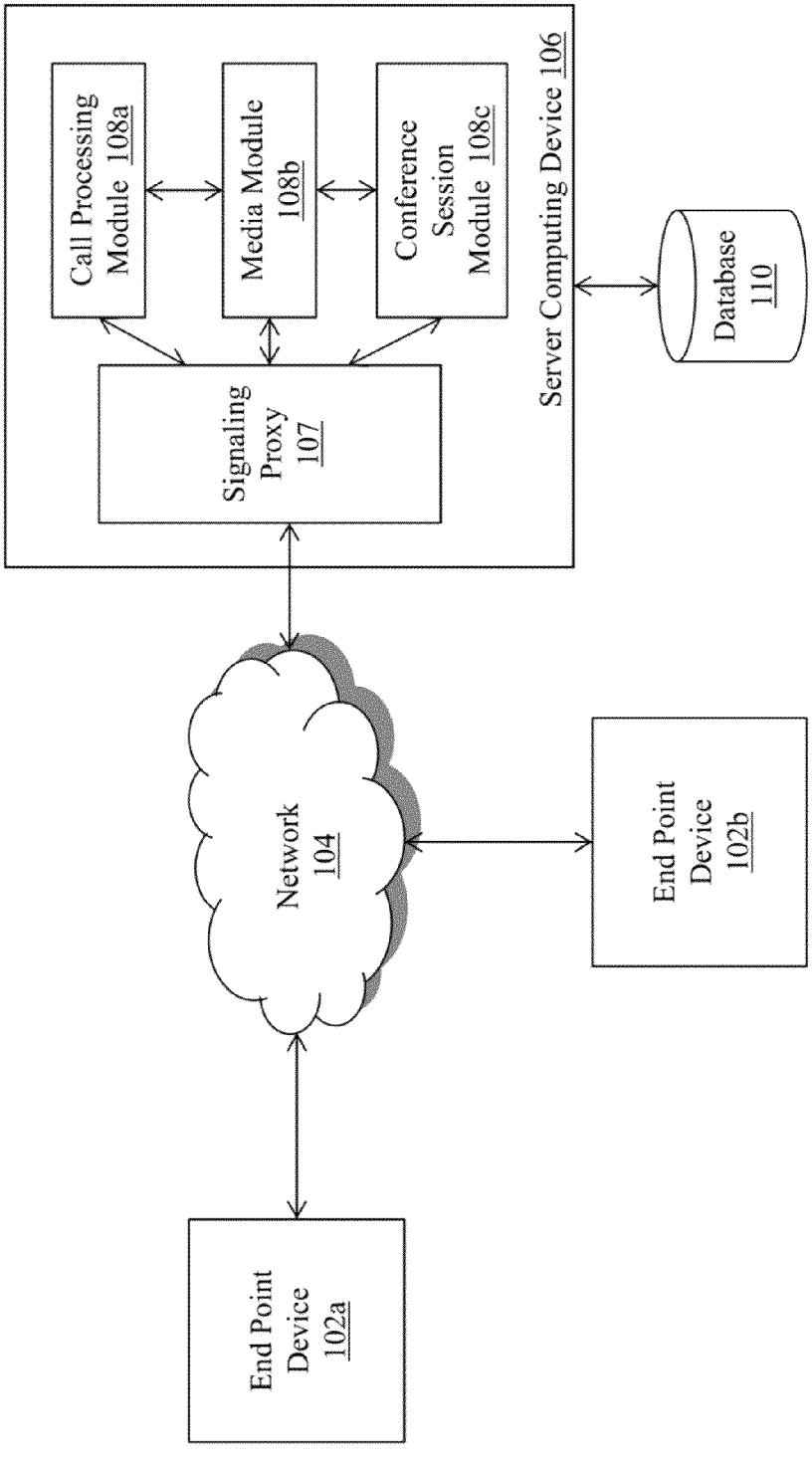
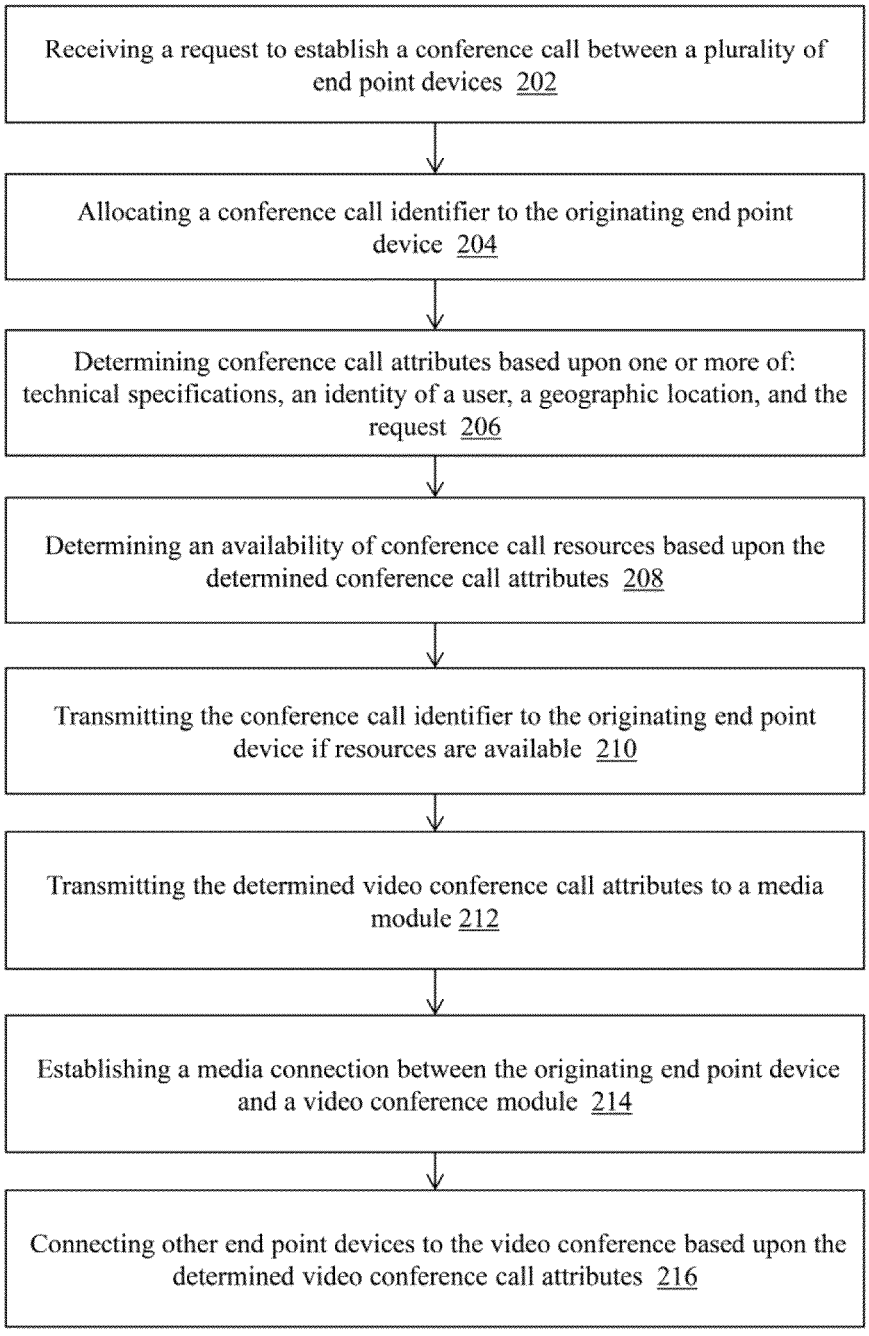

Server Computing Device 106
Call Processing Module 108a
Media Module 108b
Conference Session Module 108c
Signaling Proxy 107
Database 110
Network 104
End Point Device 102b
End Point Device 102a
100

FIG. 1


Receiving a request to establish a conference call between a plurality of end point devices 202
Allocating a conference call identifier to the originating end point device 204
Determining conference call attributes based upon one or more of: technical specifications, an identity of a user, a geographic location, and the request 206
Determining an availability of conference call resources based upon the determined conference call attributes 208
Transmitting the conference call identifier to the originating end point device if resources are available 210
Transmitting the determined video conference call attributes to a media module 212
Establishing a media connection between the originating end point device and a video conference module 214
Connecting other end point devices to the video conference based upon the determined video conference call attributes 216

200

FIG. 2

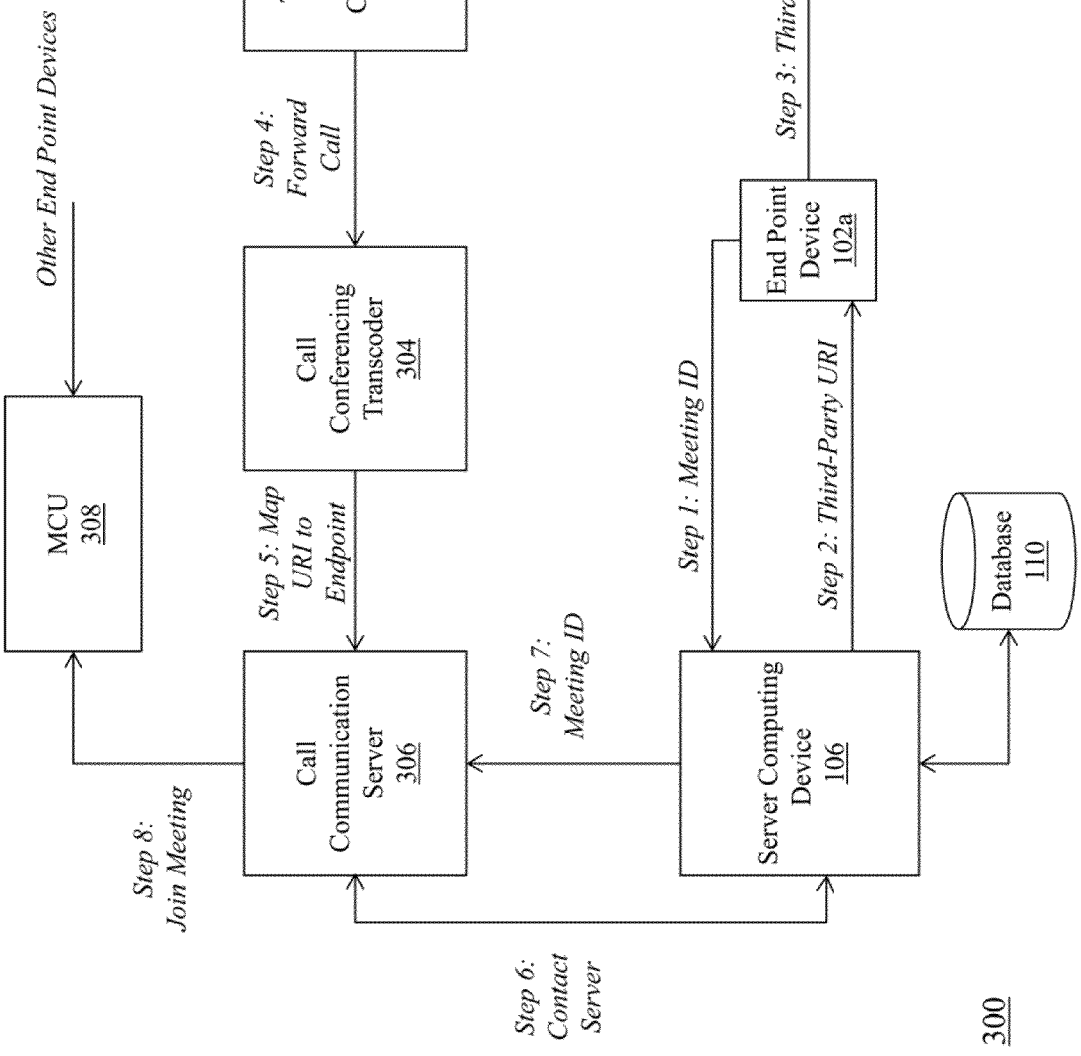

Other End Point Devices
MCU 308
Third-Party Call Conferencing Server 302
Step 4: Forward Call
Call Conferencing Transcoder 304
Step 5: Map URI to Endpoint
Step 3: Third-Party URI & ID
End Point Device 102a
Step 1: Meeting ID
Step 2: Third-Party URI
Database 110
Call Communication Server 306
Step 7: Meeting ID
Server Computing Device 106
Step 8: Join Meeting
Step 6: Contact Server
300

FIG. 3

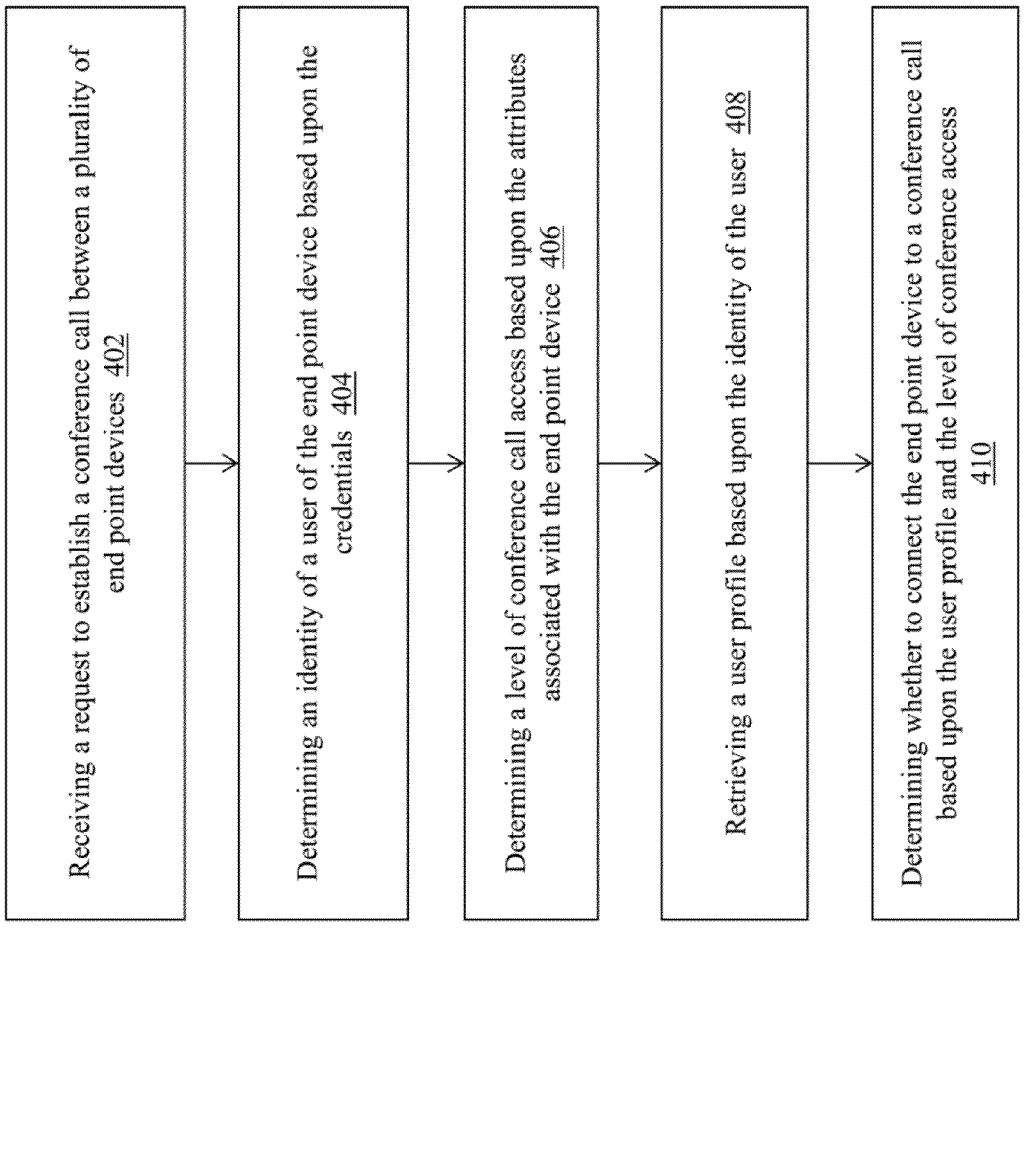

Receiving a request to establish a conference call between a plurality of end point devices 402
Determining an identity of a user of the end point device based upon the credentials 404
Determining a level of conference call access based upon the attributes associated with the end point device 406
Retrieving a user profile based upon the identity of the user 408
Determining whether to connect the end point device to a conference call based upon the user profile and the level of conference access 410
400

FIG. 4

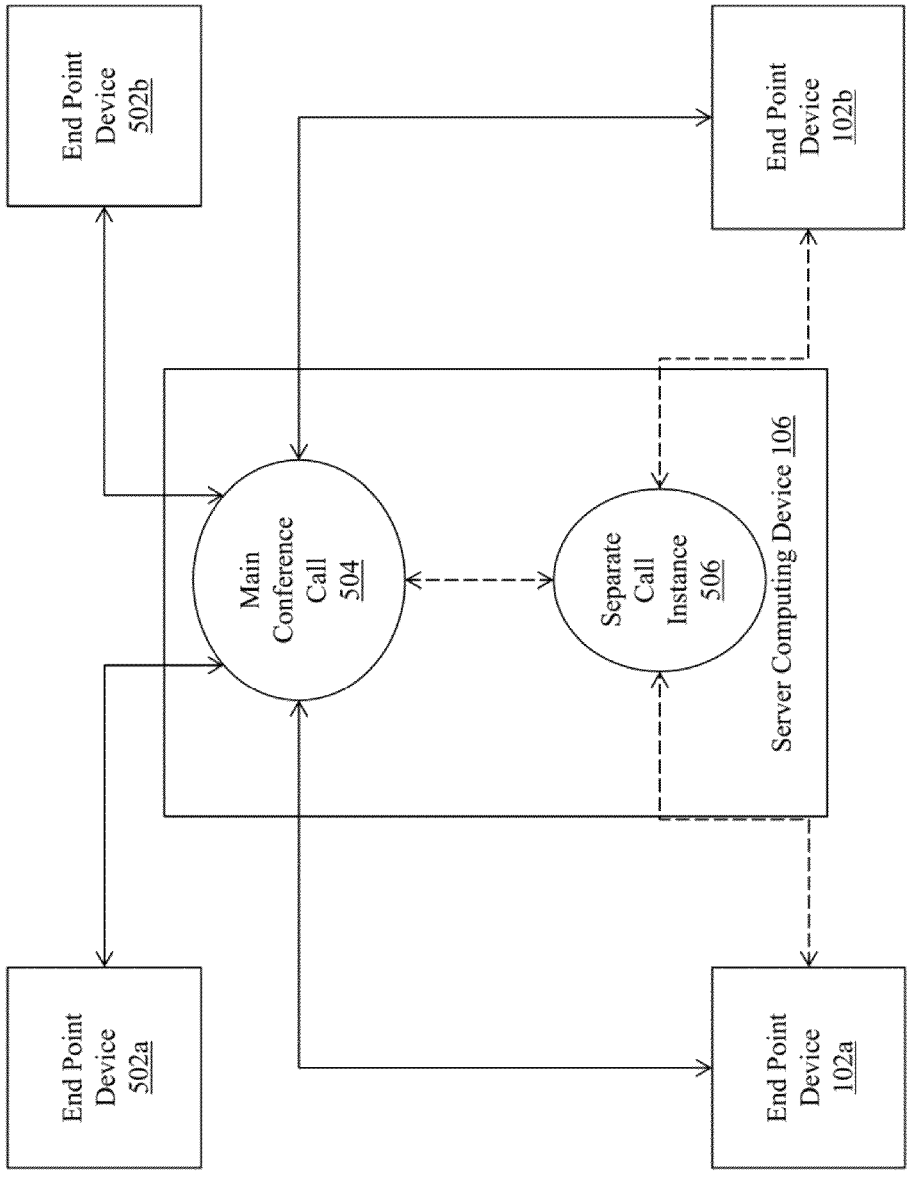

End Point Device 502b
End Point Device 102b
Main Conference Call 504
Separate Call Instance 506
Server Computing Device 106
End Point Device 502a
End Point Device 102a
500

FIG. 5

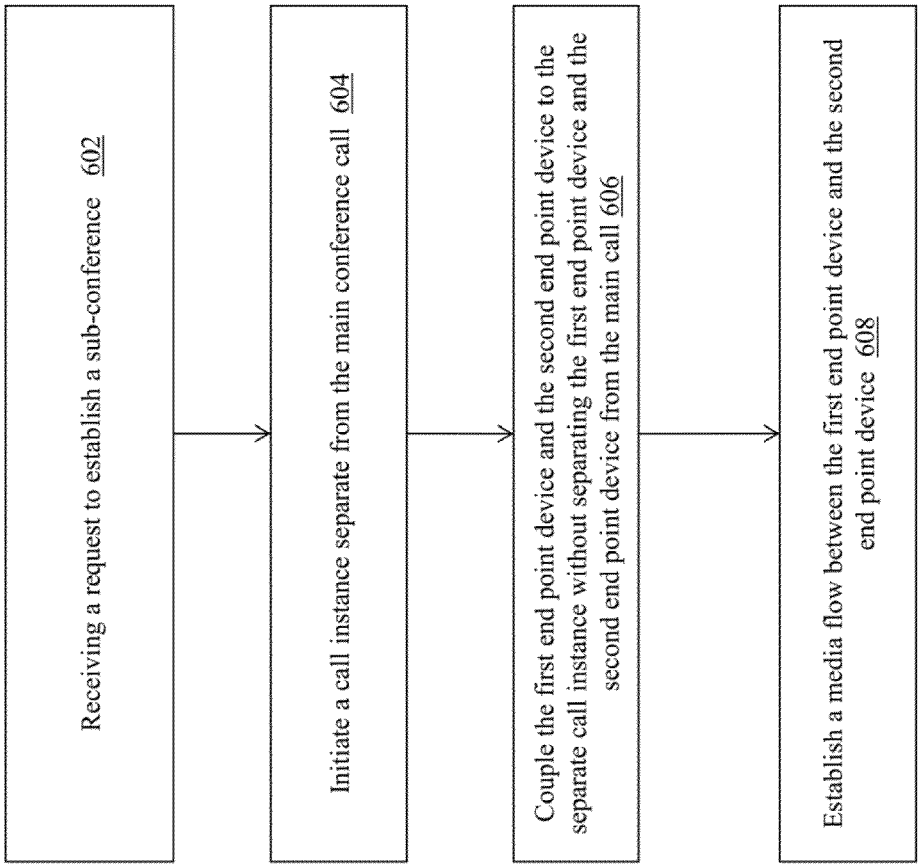

Receiving a request to establish a sub-conference 602
Initiate a call instance separate from the main conference call 604
Couple the first end point device and the second end point device to the separate call instance without separating the first end point device and the second end point device from the main call 606
Establish a media flow between the first end point device and the second end point device 608
600

FIG. 6

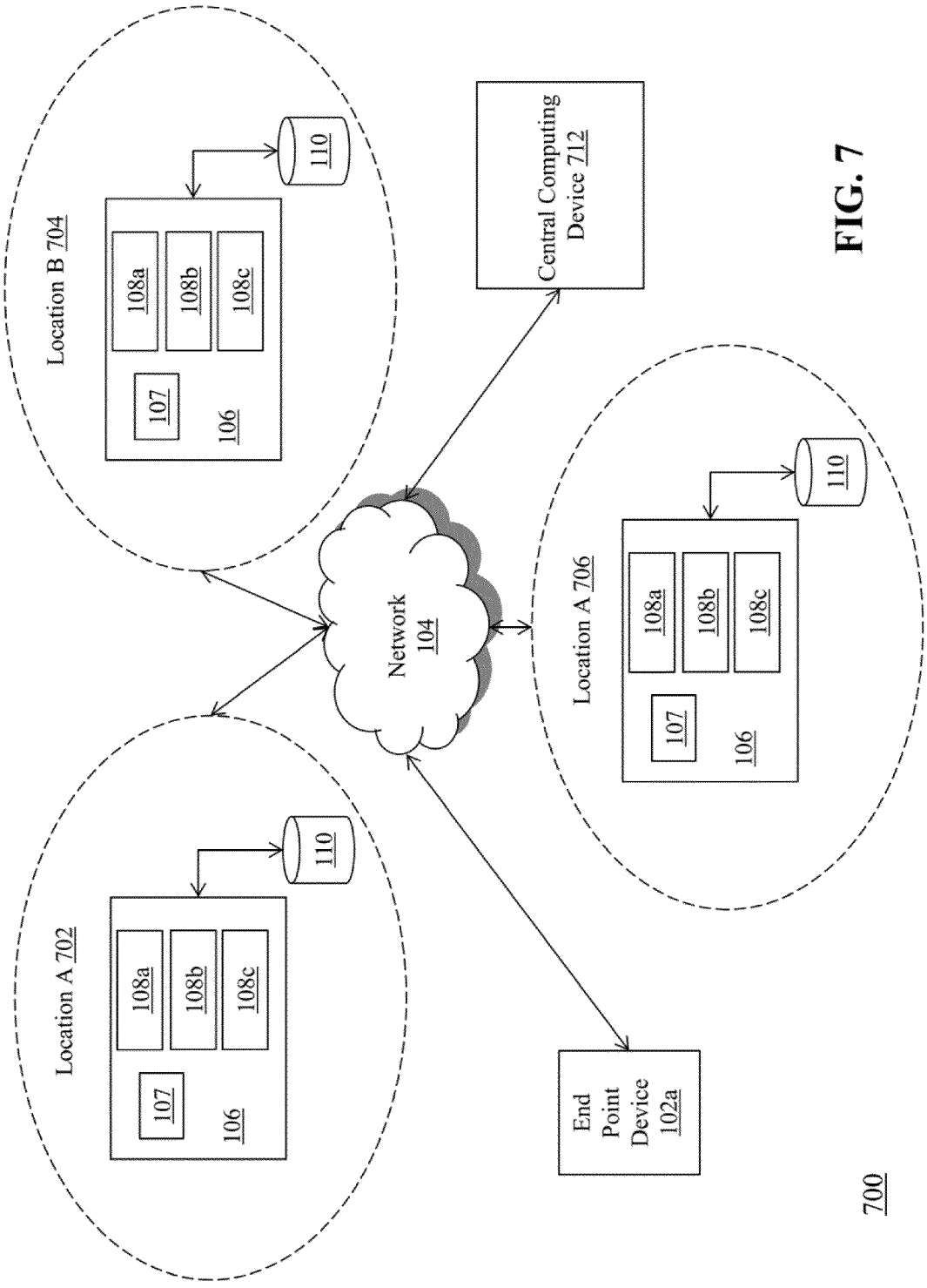

Location B 704
110
108a
108b
108c
107
106
Central Computing Device 712
Network 104
Location A 706
110
108a
108b
108c
107
106
Location A 702
110
108a
108b
108c
107
106
End Point Device 102a
700

FIG. 7

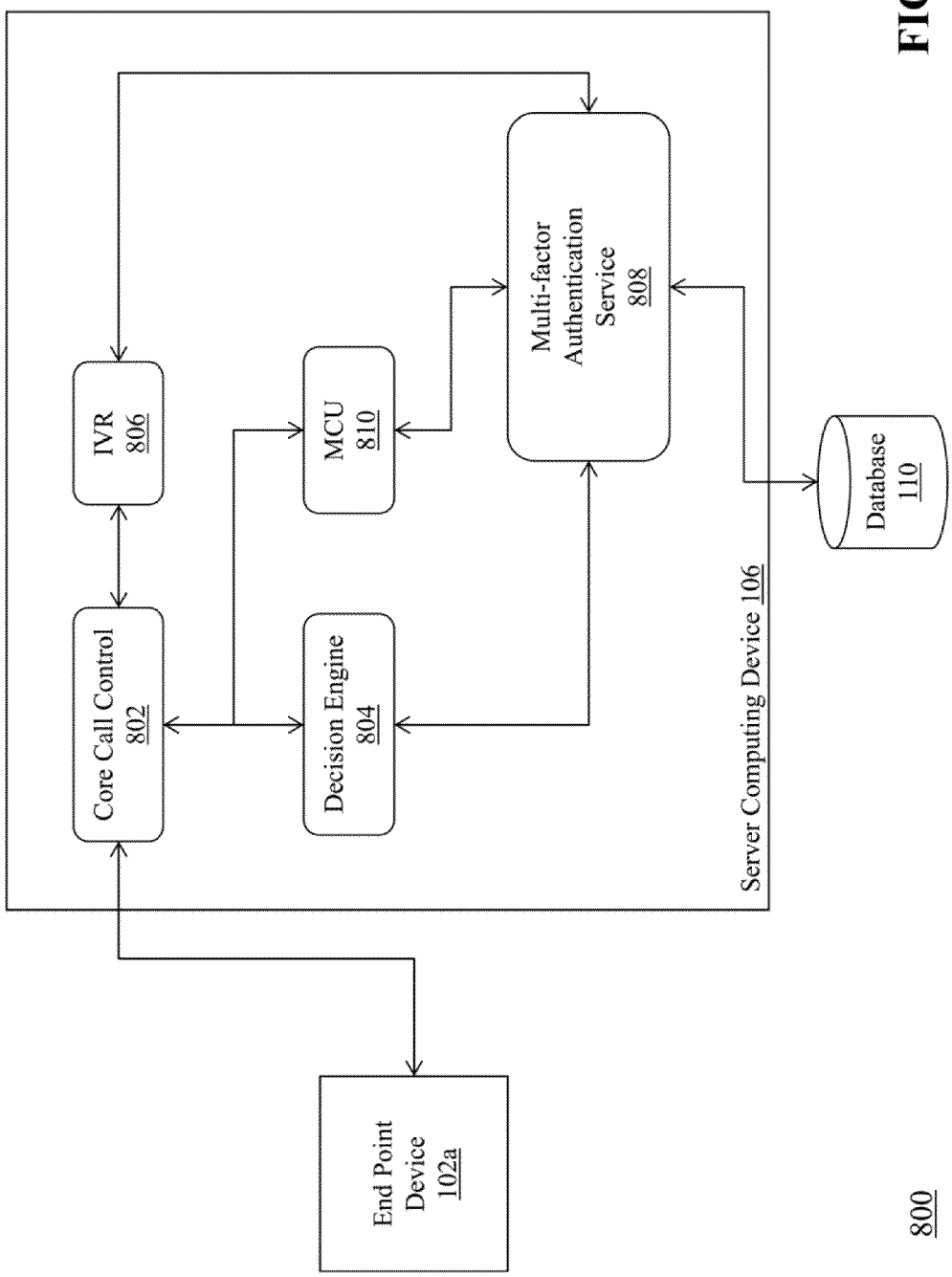

Server Computing Device 106
Core Call Control 802
IVR 806
Decision Engine 804
MCU 810
Multi-factor Authentication Service 808
Database 110
End Point Device 102a
800

FIG. 8

METHODS AND SYSTEMS FOR MULTI-FACTOR AUTHENTICATION IN SECURE MEDIA-BASED CONFERENCING

RELATED APPLICATIONS

This application is a continuation of U.S. patent application Ser. No. 14/480,091, filed Sep. 8, 2014, which is a continuation-in-part of U.S. patent application Ser. No. 14/289,122, filed on May 28, 2014, now U.S. Pat. No. 8,970,660, which claims priority to U.S. Provisional Patent Application No. 61/889,926, filed on Oct. 11, 2013, all of which are incorporated herein in their entirety.

TECHNICAL FIELD

This application relates generally to methods and apparatuses, including computer program products, for multi-factor authentication in secure media-based conferencing.

BACKGROUND

Media-based (i.e., video and/or audio) conferencing has traditionally lacked security, call enhancement, and interoperability features. Typically, conference calls occur over private networks between end points that are known to the network and call conferencing system, and have already been authenticated. In some cases, these private networks operate on a proprietary software and/or hardware platform from a specific vendor, making it difficult for end points outside of the scope of the proprietary technology to access the call conferencing platform. In addition, typical conference call signaling is exchanged directly between the end points, which makes security of the signaling harder to achieve.

SUMMARY

Therefore, what is needed are methods and systems to conduct secure media-based conferencing while offering a robust feature set that provides call enhancement features such as interactive voice response (IVR) functionality and auto attendance, call security features such as password management, multi-factor authentication and authorization of end points (including filtering and allow/deny functionality), and call compliance features such as recording options, regulatory rules, and other retention/surveillance features. The systems and methods described herein provide the advantage of interrupting the signaling associated with a media-based conference call to provide dynamic decision-making with respect to routing, permissions, authentication, and the like. The systems and methods described herein provide the advantage of secure multi-factor authentication of end points connecting to the call conferencing system. The systems and methods described herein provide the advantage of media-based conferencing with seamless interoperability between end points operating on varying types of software and hardware platforms.

The systems and methods described herein provide the advantage of dynamically allocating hardware and software resources of the system to ensure availability and efficient routing of media-based conference calls. For example, the hardware and software resources allocated by the systems and methods described herein can be resident on a plurality of geographically-dispersed and independent nodes (e.g., not in the same physical area) that communicate via a mesh-based framework. This attribute enables the system to provide the advantage of a componentized call system, instead of the traditional 'single-box' systems commonly used. Further, the processes and modules that comprise the system can operate independently of each other, without relying on other nodes or modules of the system, in making decisions about which actions to take with respect to a given conference call, end point device, or user. The systems and methods described herein achieve the advantage of disassociating the 'conference call' from any specific physical infrastructure.

The systems and methods described herein provide the further advantage of enabling private sub-conferences between participants to a main conference call while applying each of the authentication, enhancement, and regulatory features set forth above.

The invention, in one aspect, features a computerized method for authenticating an end point device participating in a media-based conference call. A call processing module of a server computing device receives a request to join a conference call between a plurality of end point devices. The request includes a media stream associated with the first end point device. The call processing module analyzes the media stream using a matrix of authentication factors, where the authentication factors include a) user-specific factors comprising one or more of: facial recognition attributes, audio recognition attributes, and user gesture attributes; b) environment factors comprising one or more of: acoustic environment attributes and visual environment attributes; and c) technical factors comprising one or more of: technical attributes of the end point device and technical attributes of the media stream. The call processing module determines an authentication score for the first end point device based upon the media stream analysis and determines whether to connect the first end point device to the conference call, another media resource, or another user of a network or communication system based upon the authentication score.

The invention, in another aspect, features a system for authenticating an end point device participating in a media-based conference call. The system includes a server computing device having a call processing module configured to receive a request to join a conference call between a plurality of end point devices. The request includes a media stream associated with the first end point device. The call processing module is further configured to analyze the media stream using a matrix of authentication factors, where the authentication factors include a) user-specific factors comprising one or more of: facial recognition attributes, audio recognition attributes, and user gesture attributes; b) environment factors comprising one or more of: acoustic environment attributes and visual environment attributes; and c) technical factors comprising one or more of: technical attributes of the end point device and technical attributes of the media stream. The call processing module is further configured to determine an authentication score for the first end point device based upon the media stream analysis and determine whether to connect the first end point device to the conference call, another media resource, or another user of a network or communication system based upon the authentication score.

The invention, in another aspect, features a computer program product, tangibly embodied in a non-transitory computer readable storage device, for authenticating an end point device participating in a media-based conference call. The computer program product includes instructions operable to cause a call processing module of a server computing device to receive a request to join a conference call between a plurality of end point devices. The request includes a media stream associated with the first end point device. The computer program product includes further instructions operable to cause the call processing module to analyze the media

stream using a matrix of authentication factors, where the authentication factors include a) user-specific factors comprising one or more of: facial recognition attributes, audio recognition attributes, and user gesture attributes; b) environment factors comprising one or more of: acoustic environment attributes and visual environment attributes; and c) technical factors comprising one or more of: technical attributes of the end point device and technical attributes of the media stream. The computer program product includes further instructions operable to cause the call processing module to determine an authentication score for the first end point device based upon the media stream analysis and determine whether to connect the first end point device to the conference call, another media resource, or another user of a network or communication system based upon the authentication score.

Any of the above aspects can include one or more of the following features. In some embodiments, the call processing module determines an identity of a user of the first end point device based upon the media stream analysis and retrieves a user profile based upon the identity of the user, the user profile including a set of permissions associated with conference call functionality. In some embodiments, the acoustic environment attributes include (i) one or more pre-recorded audio files from a user and (ii) acoustic attributes of a room in which the first end point device is located. In some embodiments, the visual environment attributes include (i) user-provided or system-captured still images and multiple video frames and (ii) lighting attributes of a room in which the first end point device is located.

In some embodiments, the technical attributes of the first end point device include an image resolution of a camera coupled to the first end point device, a device identifier, a location, and an originating address. In some embodiments, the technical attributes of the media stream include a media format and a media quality.

In some embodiments, the call processing module transmits the media stream associated with the first end point device to one or more other end point devices connected to the conference call, receives a validation signal from the one or more other end point devices to confirm an identity of a user of the first end point device, and adds the received validation signal to the matrix of authentication factors. In some embodiments, the media stream analysis includes comparing the matrix of authentication factors for the media stream to a matrix of authentication factors for prior media streams associated with the first end point device. In some embodiments, the media stream analysis includes analyzing the matrix of authentication factors for the media stream based upon user- and/or system-specified preferences.

In some embodiments, the call processing module stores the media stream analysis and authentication score for the media stream in a database. In some embodiments, the media stream analysis and authentication score determination occur periodically during the conference call. In some embodiments, where the authentication score is below a predetermined threshold, the call processing module transmits a request for user credentials to the first end point device. In some embodiments, the call processing module transmits the authentication score for the first end point device for display on one or more other end point devices connected to the conference call.

In some embodiments, the call processing module denies the request to join the conference call when the authentication score is below a predetermined threshold and disconnects the first end point device. In some embodiments, the call processing module allows the request to join the conference call when the authentication score is at or above a predetermined threshold and connects the first end point device to the conference call.

Other aspects and advantages of the invention will become apparent from the following detailed description, taken in conjunction with the accompanying drawings, illustrating the principles of the invention by way of example only.

BRIEF DESCRIPTION OF THE DRAWINGS

The advantages of the invention described above, together with further advantages, may be better understood by referring to the following description taken in conjunction with the accompanying drawings. The drawings are not necessarily to scale, emphasis instead generally being placed upon illustrating the principles of the invention.

FIG. **1** is a block diagram of a system for media-based conferencing between a plurality of end point devices, according to an embodiment of the invention.

FIG. **2** is a flow diagram of a method for media-based conferencing between a plurality of end point devices, according to an embodiment of the invention.

FIG. **3** is a block diagram of a system for media-based conferencing between a plurality of end points, according to an embodiment of the invention

FIG. **4** is a flow diagram of a method for authenticating an end point device participating in a conference call, according to an embodiment of the invention.

FIG. **5** is a block diagram of a system for establishing a sub-conference between a plurality of end point devices participating in a conference call, according to an embodiment of the invention.

FIG. **6** is a flow diagram of a method for establishing a sub-conference between a plurality of end point devices participating in a conference call, according to an embodiment of the invention.

FIG. **7** is a block diagram of a networked system for media-based conferencing between a plurality of end points, according to an embodiment of the invention.

FIG. **8** is a flow diagram of a multi-factor authentication process of an end point device participating in a conference call, according to an embodiment of the invention.

DETAILED DESCRIPTION

FIG. **1** is a block diagram of a system **100** for media-based conferencing between a plurality of end point devices, according to an embodiment of the invention. The system **100** includes a plurality of end point devices **102***a*-**102***b*, a communications network **104**, a server computing device **106** that includes a signaling proxy **107**, a call processing module **108***a*, a media module **108***b*, and a conference session module **108***c*, and a database **110**.

The plurality of end point devices **102***a*-**102***b* connect to the server computing device **106** via the communications network **104** in order to initiate and participate in conference calls and other media communication sessions with other end points. Exemplary end point devices include desktop computers, laptop computers, tablets, mobile devices, smartphones, and internet appliances. It should be appreciated that other types of computing devices that are capable of connecting to the server computing device **106** can be used without departing from the scope of invention. In some embodiments, the end point devices **102***a*-**102***b* are capable of executing call conferencing client software locally and/or using another type of user interface (e.g., a web browser) to connect to the server computing device **106**. The call conferencing client

software can be open network, free-to-use/freemium software, such as Skype™ available from Microsoft Corp. of Redmond, Wash. or Google™ Hangouts available from Google, Inc. of Mountain View, Calif., or purchasable, closed network software, such as the RealPresence® platform available from Polycom, Inc. of San Jose, Calif. In some embodiments, the call conferencing client software can be a proprietary platform developed, e.g., by a corporation for use internally. Although FIG. **1** depicts two end point devices **102***a*-**102***b*, it should be appreciated that the system **100** can include any number of end point devices.

The communication network **104** enables the end point devices **102***a*-**102***b* to communicate with the server computing device **106** in order to initiate and participate in media-based conference calls and meetings. The network **104** may be a local network, such as a LAN, or a wide area network, such as the Internet and/or a cellular network. In some embodiments, the network **104** is comprised of several discrete networks and/or sub-networks (e.g., cellular→Internet) that enable the end point devices **102***a*-**102***b* to communicate with the server computing device **106**.

The server computing device **106** is a combination of hardware and software modules that establish, authorize, facilitate and manage media-based conference calls and meetings between a plurality of end point devices **102***a*-**102***b*. The server computing device **106** includes a signaling proxy **107**, a call processing module **108***a*, a media module **108***b*, and a conference session module **108***c*. The proxy **107** and modules **108***a*-**108***c* are hardware and/or software modules that reside on the server computing device **106** to perform functions associated with establishing, authorizing, facilitating, and managing media-based conference calls and meetings. In some embodiments, the functionality of the proxy **107** and the modules **108***a*-**108***c* is distributed among a plurality of computing devices. It should be appreciated that any number of computing devices, arranged in a variety of architectures, resources, and configurations (e.g., cluster computing, virtual computing, cloud computing) can be used without departing from the scope of the invention. It should also be appreciated that, in some embodiments, the functionality of the proxy **107** and the modules **108***a*-**108***c* can be distributed such that any of the proxy **107** and/or the modules **108***a*-**108***c* are capable of performing any of the functions described herein without departing from the scope of the invention. For example, in some embodiments, the functionality of the proxy **107** and/or the modules **108***a*-**108***c* can be merged into a single module or, in some embodiments, the modules **108***a*-**108***c* can be merged into a single module coupled to the proxy **107**.

The signaling proxy **107** is coupled between the end point devices **102***a*-**102***b* and the modules **108***a*-**108***c*. The proxy **107** receives signaling communications in various protocols (e.g., Session Initiation Protocol (SIP), h.323) from the end point devices **102***a*-**102***b* that relate to the establishment and maintenance of media-based conference call sessions. It should be appreciated that other signaling protocols can be used without departing from the scope of invention. The proxy **107** receives the signaling communications from the end point devices **102***a*-**102***b* and transmits the signaling to the modules **108***a*-**108***c* for further processing.

In some embodiments, the proxy **107** translates the signaling into another format that can be processed by the modules **108***a*-**108***c*. For example, the proxy **107** can translate the signaling into XML format, including items such as call details (e.g., to, from, domain), end point device-specific details, user-specific details, and other types of information, thereby offering additional customization of the signaling to enable the modules **108***a*-**108***c* to process the call in a dynamic way for each end point device **102***a*-**102***b*.

Upon receiving the signaling from the proxy **107**, the call processing module **108***a* can perform a number of different actions to process the call. In some embodiments, the call processing module **108***a* analyzes the signaling and redirects the call to other resources in the system **100** for further processing. For example, the call processing module **108***a* can determine that the inbound call signaling is originating from an end point device that is operating a specific call conferencing hardware and/or software platform. Based upon the platform determination, the call processing module **108***a* can redirect the signaling to a resource in the system that is capable of communicating with the platform of the end point device **102***a*. In some embodiments, the call processing module **108***a* returns a response to the end point device **102***a* that originated the signaling, where the response includes call routing data (e.g., a URI) for the end point device to re-route the signaling. In some embodiments, the call processing module **108** returns a response to the proxy **107** (e.g., XML) and the proxy **107** handles the routing of the signaling session to the destination end point device.

In some embodiments, the call processing module **108***a* uses the signaling to identify a user of the originating end point device **102***a* and/or the type of end point device **102***a* that originated the signaling. For example, the call processing module **108***a* can utilize data in the signaling, such as the 'to' address, the 'from' address, a device identifier, a user ID, and the like, to determine the identity of a user associated with the originating end point device or the destination end point device. The call processing module **108***a* can access the database **110** to look up details of the user based upon any of the above data points. For example, if the signaling includes a 'to' address, the call processing module **108***a* can search in the database **110** for a user profile associated with the 'to' address. In this way, the call processing module **108***a* maps the signaling to a user and can then leverage its capabilities to customize the conference experience based upon that user's identity.

In another example, the call processing module **108***a* can use the signaling to determine the technical capability of the end point device **102***a* and adjust the conferencing features and options available to that end point device. The signaling can include a data point that indicates the originating end point device **102***a* has limited network bandwidth for sending and receiving data. The call processing module **108***a* can upgrade or downgrade the fidelity of the media transmitted to the originating end point device **102***a* based upon the available bandwidth capabilities of the device **102***a*.

In another example, the call processing module **108***a* can use the signaling to determine a user associated with the end point device (as described above) and then perform authentication of the end point device/user to determine the level of access that the user has on the system **100**. For example, the call processing module **108***a* can determine that the user is restricted from establishing media-based conference calls with a specified list of destinations (e.g., people, devices, physical locations). Based upon the determination of these restrictions, the call processing module **108***a* can evaluate whether to establish the conference call between the originating end point device **102***a* and the destination end point device specified in the signaling.

As described above, the server computing device **106** also includes a media module **108***b*. The media module is coupled to the proxy **107** and the other modules **108***a* and **108***c*. The media module **108***b* performs media signaling and streaming functionality, including acting as a call termination and

streaming point. In some embodiments, the media module **108***b* performs functions to locate an intermediate point (e.g., server device) between the participating end point devices to anchor the media and may not process the media flow. In some embodiments, once the call processing module **108***a* has established the conference call between a plurality of end point devices based upon the signaling, the call processing module **108***a* can transfer the media associated with the call to the media module **108***b* for handling the media session between the end point devices. The media module **108***b* also provides additional conference call enhancement features, such as Interactive Voice Response (IVR) menus and prompts, auto-attendants, and advanced PIN management.

In some embodiments, the media module **108***b* includes private branch exchange (PBX) software for managing the media signaling and flows of conference calls processed by the server computing device **106**. An example PBX software platform that can be incorporated into the media module is Asterisk®, available from Digium, Inc. of Huntsville, Ala.

The server computing device **106** also includes a conference session module **108***c*. The conference session module **108***c* is coupled to the proxy **107** and the other modules **108***a* and **108***b*. The conference session module **108***c* performs functions to bridge the plurality of end point devices participating in a media-based conference call or meeting into a single session. In some embodiments, the conference session module **108***c* is a multipoint control unit (MCU). An example MCU that can be incorporated into the conference session module **108***c* is Codian, available from Cisco Systems, Inc. in San Jose, Calif. The MCU can be integrated with Vidtel Gateway, available from Vidtel, Inc. of Sunnyvale, Calif., to provide additional features.

The system **100** also includes a database **110**. The database **110** is coupled to the server computing device **106** and stores data used by the server computing device **106** to perform the media-based conferencing functionality. The database **110** can be integrated with the server computing device **106** or be located on a separate computing device. An example database that can be used with the system **100** is MySQL™ available from Oracle Corp. of Redwood City, Calif.

FIG. **2** is a flow diagram of a method **200** for media-based conferencing between a plurality of end point devices, using the system **100** of FIG. **1**. The call processing module **108***a* of the server computing device **106** receives (**202**) a request to establish a conference call between a plurality of end point devices (e.g., **102***a*-**102***b*). The request originates from one of the end point devices (e.g., **102***a*). For example, in the SIP context, the originating end point device **102***a* transmits a SIP INVITE message to the server computing device **106** via the network **104**. The proxy **107** receives the SIP INVITE message and, in some embodiments, translates the SIP INVITE message into XML, which is then passed on to the call processing module **108***a*.

The call processing module **108***a* allocates (**204**) a conference call resource identifier to the originating end point device **102***a*, based upon the received XML request. As will be described in greater detail below, the call processing module **108***a* also determines the availability of resources for the conference call and transmits the conference call resource identifier to the originating end point device **102***a* only if specified and/or necessary resources are available.

The call processing module **108***a* determines (**206**) conference call attributes based upon one or more of: technical specifications of the originating end point device **102***a*, an identity of a user associated with the originating end point device **102***a*, a geographic location of the originating end point device, and the request to establish the conference call.

As described above, the call processing module **108***a* receives the request in XML format from the proxy **107** and evaluates data embedded in the request to determine processing for the conference call request. For example, the call processing module **108***a* can evaluate the origination address (e.g., 'from' address) in the request to identify a user of the end point device **102***a* in order to perform functions such as authentication, group permissions, feature access, and the like. In another example, the call processing module **108***a* can evaluate the request to determine certain technical attributes of the originating and/or destination end point devices. For instance, if the call request originates at a mobile device and/or via a cellular network, the call processing module **108***a* can determine extra and/or different call security options (e.g., encryption). In another example, the call processing module **108***a* can evaluate the geographic location of the originating end point device **102***a* to determine appropriate routing rules, security authorizations, and/or hardware/software resources to allocate to the originating end point device **102***a* or the conference call itself. It should be appreciated that the call processing module **108***a* can analyze the request to establish a conference call to determine attributes associated with the call and/or attributes that the module **108***a* should assign to the call as the request arrives.

As mentioned above, the call processing module **108***a* determines (**208**) an availability of resources based upon the conference call attributes. For example, the call processing module **108***a* determines a resource (e.g., gateway, URI) available for the requested conference call and generates a conference call resource identifier to be assigned to the originating end point device **102***a*. For example, if the originating end point device **102***a* is using a particular software platform (e.g., Skype™) to initiate the media-based conference call, the call processing module **108***a* generates a Skype™ URI to be associated with the end point device **102***a*. In some embodiments, if a particular resource is unavailable (e.g., all of the Skype™ URIs are in use), the call processing module **108***a* can inform the originating end point device **102***a* and wait until a resource is available or decline to establish the conference call.

The call processing module **108***a* can then transmit (**210**) the conference call identifier (e.g., URI) to the end point device **102***a* for redirection of the call signaling to a server associated with the software platform. In some embodiments, the conference call resource identifier is associated with resources internal to the system **100** that can handle and process the call signaling (e.g., an internal gateway, transcoder).

The call processing module **108***a* transmits (**212**) the determined conference call attributes to the media module **108***b*. For example, the call processing module **108***a* can transmit the determined conference call attributes (e.g., via XML) to the media module **108***b* for providing features such as call enhancement functionality, call routing functionality, media streaming functionality, and call termination functionality.

The media module **108***b* establishes (**214**) a conference media connection between the originating end point device **102***a* and the conference session module **108***c* for initiation of a conference media flow. In some embodiments, once the call processing module **108***a* has established the signaling connection and performed various functions (e.g., authentication, permissioning, routing), the media module **108***b* initiates a media flow between the originating end point device **102***a* and the conference session module **108***c* at the server computing device **106**. The media module **108***b* can still manage the call state and media streaming, while the conference session module **108***c* connects (**216**) the various end

point devices participating in the conference call to the conference call session. In some embodiments, the conference session module **108***c* uses the determined conference call attributes to determine whether certain end point devices are able to connect to the conference call.

FIG. **3** is a block diagram of a system **300** for media-based conferencing between a plurality of end points, based upon the system **100** of FIG. **1**. The system **300** includes the end point device **102***a*, the server computing device **106**, the database **110**, a third-party call conferencing server computing device **302**, a call conferencing transcoder **304**, a media-based communication server (MCS) computing device **306**, and a MCU **308**.

FIG. **3** depicts an example workflow for establishing a media-based conference call between a plurality of end point devices:

Step **1**: the end point device **102***a* transmits a request to establish a media-based conference call to the server computing device **106**. The request includes a meeting ID. For example, if the end point device **102***a* is operating via Skype™ client software, a user at the device **102***a* clicks a button in the Skype™ user interface to initiate the conference call. The software transmits the user's Skype™ ID and the meeting ID to the server computing device **106** to establish a signaling session with the server **106**.

Step **2**: the server computing device **106** processes the request to determine the identity of the user associated with the request. For example, the server **106** can retrieve a user ID from the request and use the database **110** to determine the identity of the user and any related information (e.g., permissions, device-specific information, and authentication details). The server can check whether the user associated with the user ID is authorized to join the meeting requested (via the meeting ID). The server computing device **106** then returns a Skype™ URI to the end point device **102***a*.

Step **3**: the end point device **102***a* uses the received URI to initiate a signaling session for the conference call with a Skype™ server (e.g., third-party call conferencing server **302**). The URI can include an address or identifier associated with the call conferencing transcoder (e.g., device **304**), the media-based communication server (e.g., device **306**), and/or the MCU **308**. In some embodiments, the URI corresponds to an MCS endpoint and the server computing device **106**/database **110** maintains a correspondence between the third-party URI and the MCS endpoint.

Step **4**: the third-party call conferencing server **302** forwards the conference call signaling to the call conferencing transcoder **304** (e.g., a Vidtel module).

Step **5**: the call conferencing transcoder **304** maps the received Skype™ URI to an MCS endpoint address. For example, the transcoder **304** can modified the user's Skype™ ID to add the MCS end point address (e.g., <user's Skype™ ID>@skype.vidtel.com). The transcoder **304** then communicates with the MCS **306**.

Step **6**: the MCS **306** communicates with the server computing device **106**. For example, the MCS **306** transmits the modified Skype™ ID that includes the MCS end point address to the server computing device **106**.

Step **7**: the server **106** uses the modified ID to locate the meeting ID that was previously transmitted to the server **106** by the end point device **102** (see Step **1**). The server **106** then transmits the meeting ID to the MCS **306**.

Step **8**: the MCS **306** then transfers the conference call signaling for the end point device **102***a* to the MCU **308** so that the user can join the media-based conference call or meeting, as requested.

It should be appreciated that FIG. **3** represents an exemplary embodiment for establishing a media-based conference call between a plurality of end point devices. Other techniques and workflows for establishing a media-based conference call between a plurality of end point devices can be contemplated without departing from the scope of invention.

In some embodiments, the system **100** is capable of providing interoperability between different types of end point devices that request to join the same media-based conference call. For example, an end point device **102***a* may access the system **100** via a Skype™ user interface over a cellular network while another end point device **102***b* may access the system via a call conferencing terminal installed behind a VoIP architecture, e.g., at a company. The server computing device **106** can perform transcoding between the respective end point devices to enable a seamless communication session. In some embodiments, the server computing device **106** reserves system resources depending on the type of request and/or end point device that is accessing the system. The server computing device **106** can maintain a pool of access so that system resources are reserved and released as appropriate based upon the initiation and termination of media-based conference calls and meetings.

FIG. **4** is a flow diagram of a method for authenticating an end point device participating in a media-based conference call, using the system **100** of FIG. **1**. The call processing module **108***a* of the server computing device **106** receives (**402**) receives a request to establish a conference call between a plurality of end point devices (e.g., end point devices **102***a*-**102***b*), the request including credentials associated with a user of an end point device (e.g., **102***a*) and attributes associated with the end point device **102***a*. The call processing module **108***a* determines (**404**) an identity of the user of the end point device based upon the credentials. The call processing module **108***a* determines (**406**) a level of conference call access based upon the attributes associated with the end point device. The call processing module **108***a* retrieves (**408**) a user profile based upon the identity of the user, the user profile including a set of permissions associated with conference call functionality. The call processing module **108***a* determines (**410**) whether to connect the end point device to a conference call based upon the user profile and the level of conference call access.

For example, the server computing device **106** can perform multi-factor authentication to determine (i) the identity of both the user and end point device that has initiated the request to establish a conference call and (ii) how the server **106** should handle the request (e.g., connect the call, route the call to another resource in the system for further processing, authenticate the user via additional means). In one example, the server computing device **106** can fingerprint the incoming request by evaluating attributes relating to a user of the end point device, the end point device itself, and/or the parameters of the requested call to create a matrix of the attributes that comprise the fingerprint. The server **106** can store the fingerprint, e.g., in the database **110** so that the fingerprint can be referenced in the future—both in comparison with requests from the same user/end point device and from different users/end point devices. The server computing device **106** can also re-evaluate the fingerprint as a conference call proceeds after establishment. For example, if the end point device is attempting to access the conference call system in an unauthorized manner by providing a temporary, fake alias to the server **106** upon transmitting the call request, but later during the call the end point device attributes change, the server **106** can dynamically and automatically evaluate the fingerprint

again and take corrective action (e.g., disconnecting the call, routing the call to another resource for monitoring) if necessary.

In some embodiments, the multi-factor authentication is implemented according to two facets: (1) technically (i.e., how/why is the end point device calling?) and (2) personally (i.e., who is the user at the end point device?). The server computing device **106** can receive a request for a media-based conference from an end point device and determine items such as technical features of the calling device (e.g., hardware, software, location, originating network, protocol) and personal features of the calling device and/or the user associated with the device (e.g., user name, ID, PIN, facial recognition). The server computing device **106** can evaluate both sets of information to provide a robust authentication process that ensures appropriate access and permissions are granted to the end point device.

The system **100** can perform multi-factor authentication based upon a matrix of user-specific authentication factors, including but not limited to: facial recognition attributes; audio recognition (e.g., speech patterns/wavelengths) attributes; acoustic environment attributes; visual environment attributes; user gesture attributes; technical attributes of the end point device being operated by a user; and technical attributes of the media stream received from the end point device. Exemplary acoustic environment attributes include acoustics of the room/environment in which the user is located during the call and acoustic attributes of the equipment (e.g., microphone) or device used to capture audio during the call. Exemplary visual environment attributes include objects in the environment around the user (e.g., walls, windows, paintings, and furniture), colors of the environment, lighting hues of the environment, and the like.

Exemplary user gesture attributes include placement and/or movement of the user's body during the call (e.g., the user's hand movement while speaking), a predetermined gesture pattern performed by the user at the beginning of the call, and other similar types of gesture recognition attributes. Exemplary technical attributes of the end point device being operated by a user include the hardware and/or software platform from which the user is connecting to the call, attributes of the image-capturing device, and address/identification attributes of a device being operated by the user (e.g., IP address, MAC address, geolocation/GPS characteristics, and the like). Exemplary technical attributes of the media stream received from the end point device include bit rate, jitter, delay, compression protocol/media format/codec utilized, transport protocol, encryption standard, and the like.

The system **100** can analyze any of the above-described authentication factors to determine an authentication score for the end point device connecting to the call. The authentication score represents a level of confidence that the end point device and/or user requesting to join a call are indeed authorized to do so. In some embodiments, the authentication score is determined by comparing the current matrix of authentication factors against historical authentication factors associated with the end point device (e.g., stored in a database) in order for the system to determine whether or not to connect the end point device to the call. In some cases, the system **100** can generate a threshold value based upon the historical authentication factors. For example, if the authentication score falls below the threshold, the system **100** can decide that the end point device is unauthorized and prevent the device from connecting to the call (and/or disconnect the device from the system **100**).

In another example, if the authentication score falls below the threshold, the system **100** can allow the end point device to connect to the call and also alert other users on the call that the end point device/user is unauthenticated. The other users may be able to authenticate the end point device/user by validating the identity of the user (e.g., using a validation signal) and informing the system **100** (i) that the user is authorized to participate and/or (ii) instructing the system **100** of the user's identity. In the latter case, the system **100** is capable of storing the current authentication factors in the user's profile (e.g., as a secondary set of authentication factors)—such that if the user/end point device requests to join a future call, the system **100** can recognize the user/end point device based upon the secondary set of authentication factors as part of a self-learning process. In some cases, the system **100** can merge the secondary set of authentication factors with authentication factors already stored for the user/end point device to result in an updated user profile.

The system **100** can display a user's/end point device's authentication score on the call screen to the other users so everyone is aware of the others' scores. In one embodiment, the score is displayed as a percent-based number (e.g., 75%) and can be displayed using color coding. For example, authentication scores above 75% are colored green, authentication scores between 50 and 75% are colored orange, and authentication scores below 50% are colored red.

In some embodiments, the above-described multi-factor authentication techniques are performed at the initiation of a call (or a request to join a call) and periodically during the call. The authentication that occurs during the call can be pre-scheduled (e.g., every fifteen minutes) or conducted dynamically based upon changes to the matrix of authentication factors. For example, if an end point device participating in a call changes its IP address/location (e.g., due to a cell phone moving geographically), the system **100** can flag that end point device for a real-time authentication. In another example, a user may pull a shade down in the room to reduce glare, thereby changing the lighting hues in the user's video. The system **100** can perform a real-time authentication of the end point device to determine whether the device is still authorized to participate in the call.

FIG. **8** is a flow diagram of a multi-factor authentication process of an end point device participating in a conference call. It should be appreciated that any of the process functions **802**, **804**, **806**, **808**, **810**_described in FIG. **8** can be performed by one or more of the modules **107**, **108***a*-**108***c* described in the system **100** of FIG. **1**.

An end point device **102***a* operated by a user transmits a request to the server computing device **106** to join an in-progress video conference call. The call from end point device **102***a* is received by the server computing device **106** at the core call control function **802**. The core call control function **802** communicates with the decision engine **804** to determine that the video conference which the end point device wants to join is a call that requires multi-factor authentication. The core call control function **802** communicates with the interactive voice response (IVR) function **806** which asks the user at the end point device **102***a* to wait while authentication is performed. The matrix of authentication factors about the end point device **102***a* (as described previously) is transmitted to the multi-factor authentication service **808** which performs the analysis described above to generate an authentication score for the end point device **102***a*. If the multi-factor authentication service **808** determines that the end point device **102***a* is authorized to join the call, the core call control function **802** transfers the media stream of the end point device **102***a* to a multi-party conferencing unit (i.e., MCU **810**) on which the video conference call is taking place. In some embodiments, the core call control function **802** can transmit the end point

device **102***a* to a waiting room (e.g., if further authentication is required, such as a passcode, or if the organizer has not yet joined). When the end point device **102***a* joins the video conference call, the MCU **810** can transmit data relating to the end point device **102***a* and its media stream to the multi-factor authentication service **808** in order to conduct the periodic/dynamic authentication procedure and the self-learning procedure described earlier. This data can be stored in a database **110** for future use.

The system **100** is also capable of performing authentication both at an individual permission level and a call/meeting permission level. For example, the server computing device **106** can retrieve individual permissions from the database **110**, e.g., based upon a Lightweight Directory Access Protocol (LDAP) or Active Directory (AD) service. The server computing device **106** can tie in to an organization's internal user profile and directory service to use the user information for managing call conferencing permissions. The server computing device **106** can also maintain and manage call/meeting permissions, such as generating or storing a list of participants that are eligible to join a particular conference call or meeting, e.g., based upon user identifier, job title, access level, or other such parameters as may be defined in relation to the user.

The system **100** has a robust permissioning scheme that allows for customized authentication and permissions at a functional level, a personal level, a group level, and/or a device level. For the functional permissions, the system **100** can determine that a particular user/end point device/conference call is able to execute certain functions, e.g., set up a sub-conference, establish a conference call, invite participants, implement security attributes, and so forth. For the personal permissions, the system **100** enables a granularity of access controls for a particular user. For example, the system **100** can determine that a user has permission to initiate a conference call of a particular type, from a particular location (e.g., work but not home), to particular people, and the like.

For the group permissions, the system **100** can assign attributes to a plurality of users based upon any number of different requirements or categorizations. For example, all of the employees in a corporation can be assigned to a group. The system **100** can also assign users to multiple groups and/or sub-groups, such as business units within the corporation or product teams within the business unit. Users can also form their own ad-hoc groups (e.g., friends, business contacts) comprising any number of users, e.g., from two users and up. It should be noted that the users do not have to be otherwise affiliated (e.g., by employer) in order to be in a group. For the device permissions, the system **100** can determine a level of access and functionality for particular call environments (e.g., hardware and/or software platform (or other technical attributes) of an end point device, location of an end point device, and so forth).

Another aspect of the permissioning structure described herein is the hierarchical nature of the permissions. For example, a corporation may assign all of its employees to a particular group with specifically-designated permission levels. Certain employees within that group may form a sub-group and want a permission level that differs from the corporation-wide permission structure. The system **100** can determine that the corporation-wide permission structure overrides the sub-group permission level, and enforce limits on the sub-group permission level so as to not exceed the scope of or conflict with the corporate-wide permissions structure.

In some scenarios, participants to a main conference call may wish to separate themselves from the conference for a brief period to communicate regarding private matters. For example, if the main conference call involves business negotiations or legal proceedings with multiple parties, one party may want to confer regarding sensitive or privileged issues without disconnecting from the main conference but still engaging in a private conversation. Therefore, the system offers a function to establish a sub-conference between multiple end point devices, and the system can apply all of the functionality described above (e.g., authentication, routing, and permissioning) to the sub-conference.

FIG. **5** is a block diagram of a system **500** for establishing a sub-conference between a plurality of end point devices (e.g., end point devices **102***a*, **102***b*, **602***a*, **602***b*) participating in a media-based conference call, based upon the system of FIG. **1**. The system **500** includes end point devices **102***a*, **102***b*, **502***a*, **502***b*, and server computing device **106**. The end point devices **102***a*, **102***b*, **502***a*, **502***b* connect to the main conference call **504** using the techniques described above. Once the main conference call **504** is established, certain participants may wish to separate themselves from the main call **504** for a period of time.

FIG. **6** is a flow diagram of a method **600** for establishing a sub-conference between a plurality of end point devices participating in a media-based conference call, using the system **100** of FIG. **1** and the system **500** of FIG. **5**. The server computing device **106** receives (**602**) a request to establish a sub-conference from a first end point device (e.g., end point device **102***a*) participating in a main conference call. The request includes an identifier associated with a second end point device (e.g., end point device **102***b*) participating in the main conference call. For example, the user at end point device **102***a* may click on an icon associated with a user at end point device **102***b* to initiate a sub-conference with the user at end point device **102***b*, and the device **102***a* transmits a request to the server computing device **106**.

The server computing device **106** initiates (**604**) a call instance (e.g., separate call instance **506**) that is separate from the main conference call **504** based upon the request to establish the sub-conference. For example, the server computing device **106** can initiate separate call instance **506** by allocating another meeting that is managed by the server **106** but where the media flow is separate from the main conference call **504**. In some embodiments, only a portion of the media flow is transferred to the separate call instance **506**. For example, the participants in the main conference call may continue to see video associated with the users involved in the sub-conference, but the audio communication exchanged between the users involved in the sub-conference is excluded from the main conference call **504**. In another example, the participants in the main conference call may hear a tone indicating that some users have initiated a sub-conference. In some embodiments, the sub-conference include the exchange of textual information (e.g., chat) between the participants in the sub-conference.

In some embodiments, the server computing device **106** changes presence information associated with the users and/or end point devices that joined the sub-conference. For example, the server computing device **106** can add a status indicator to a user's name or picture in a video conference indicating to all participants in the main conference call that the user is in a sub-conference. In another example, the server computing device **106** can periodically play an audio message to participants in the main conference call that certain users have entered a sub-conference and identify those users by name.

The server computing device **106** couples (**606**) the first end point device **102***a* and the second end point device **102***b*

to the separate call instance **506** without separating the first end point device **102***a* and the second end point device **102***b* from the main conference call **504**. The server computing device **106** establishes (**608**) a media flow between the first end point device **102***a* and the second end point device **102***b*, where the main conference call **504** is prevented from accessing the media flow between the first end point device **102***a* and the second end point device **102***b*.

The server computing device **106** can apply any of the concepts described above with respect to call routing, enhancement features, permissions, authentication, and the like to the sub-conference call instance **506**. For example, if a participant in the main conference call seeks to establish a sub-conference with another participant, the server **106** can determine whether the first participant is allowed to communicate privately with the second participant, e.g., via a set of rules stored in the database **110**. The rules can be pre-determined in advance of the main conference call or agreed upon by the participants at the time of the call. The rules can change during the course of the call, e.g., as participants come and go from the main conference call. The permissions and related features can be hierarchical across the main conference call and the sub-conference, so that a user has the same permission scope in each call instance, or in some embodiments, a user may have reduced permission scope in a sub-conference versus his permission scope in a main conference call.

It should be noted that, in some embodiments, the server computing device **106** keeps track of the status of each participant in the conference call. For example, the server **106** records certain data about the circumstances in which the users entered the sub-conference (e.g., time at which the sub-conference was initiated, identify of users in the sub-conference, how long the sub-conference lasted, and the like). This feature allows for detailed reporting about the structure and timing of the conference call, which is useful for audit and compliance purposes.

In certain contexts, the rules provide for certain surveillance and reporting features to be applied to the sub-conferences. For example, an industry may be regulated such that all communications must be recorded and documented according to specific guidelines. The server computing device **106** can determine, based upon the rules, whether to record the sub-conferences, identify the participants to the sub-conferences, and other similar requirements. In some cases, the rules applied to the main conference call are automatically transferred to the sub-conferences. In addition, the system can provide additional features such as surveillance or reporting to be used in conjunction with the sub-conference.

In addition, an advantage provided by the methods and systems described herein is that each end point device (e.g., device **102***a* of FIG. **1**) has its own media stream when participating in a conference call. Accordingly, the server computing device **106** and/or other intermediary servers and devices can insert artifacts that are personalized or specific to the user of a certain end point device (e.g., notifications, alerts, messages, graphics) without inserting the artifacts into the media streams of other end point devices. This technique enables the system **100** to perform dynamic and personalized user interface, recording, and display operations for individual users.

The techniques may be implemented in a networked system **700** comprising multiple computing devices distributed across different locations, as shown in FIG. **7**. Each of Location A **702**, Location B **704** and Location C **706** includes the server computing device **106** having components **107**, **108***a*-**108***c*, and **110** of FIG. **1**, and the servers at locations **702**, **704**, and **706** are connected to each other via the network **104**. The networked system of FIG. **7** enables distribution of the processing functions described herein across several computing devices and provides redundancy in the event that a computing device at one location is offline or inoperable. In some embodiments, end point devices (e.g., device **102***a*) in proximity to a particular location (e.g., Location A **702**) access the networked system via the server **106** at that location. In some embodiments, the server computing devices **106** at the respective locations **702**, **704**, **706** communicate with a central computing device **712** (e.g., a server) that is coupled to the network. The central computing device **712** can provide data and/or processing resources for the network of computing devices **106** (e.g., synchronization of functionality/data across the computing devices).

It should be understood that any of the above-described methods, systems, and techniques can be implemented in the context of video conferencing (i.e., conference calls consisting of video and audio media) and audio-only conferencing without departing from the scope of invention.

The above-described techniques can be implemented in digital and/or analog electronic circuitry, or in computer hardware, firmware, software, or in combinations of them. The implementation can be as a computer program product, i.e., a computer program tangibly embodied in a machine-readable storage device, for execution by, or to control the operation of, a data processing apparatus, e.g., a programmable processor, a computer, and/or multiple computers. A computer program can be written in any form of computer or programming language, including source code, compiled code, interpreted code and/or machine code, and the computer program can be deployed in any form, including as a stand-alone program or as a subroutine, element, or other unit suitable for use in a computing environment. A computer program can be deployed to be executed on one computer or on multiple computers at one or more sites.

Method steps can be performed by one or more processors executing a computer program to perform functions of the invention by operating on input data and/or generating output data. Method steps can also be performed by, and an apparatus can be implemented as, special purpose logic circuitry, e.g., a FPGA (field programmable gate array), a FPAA (field-programmable analog array), a CPLD (complex programmable logic device), a PSoC (Programmable System-on-Chip), ASIP (application-specific instruction-set processor), or an ASIC (application-specific integrated circuit), or the like. Subroutines can refer to portions of the stored computer program and/or the processor, and/or the special circuitry that implement one or more functions.

Processors suitable for the execution of a computer program include, by way of example, both general and special purpose microprocessors, and any one or more processors of any kind of digital or analog computer. Generally, a processor receives instructions and data from a read-only memory or a random access memory or both. The essential elements of a computer are a processor for executing instructions and one or more memory devices for storing instructions and/or data. Memory devices, such as a cache, can be used to temporarily store data. Memory devices can also be used for long-term data storage. Generally, a computer also includes, or is operatively coupled to receive data from or transfer data to, or both, one or more mass storage devices for storing data, e.g., magnetic, magneto-optical disks, or optical disks. A computer can also be operatively coupled to a communications network in order to receive instructions and/or data from the network and/or to transfer instructions and/or data to the network. Computer-readable storage mediums suitable for embodying computer program instructions and data include all forms of

volatile and non-volatile memory, including by way of example semiconductor memory devices, e.g., DRAM, SRAM, EPROM, EEPROM, and flash memory devices; magnetic disks, e.g., internal hard disks or removable disks; magneto-optical disks; and optical disks, e.g., CD, DVD, HD-DVD, and Blu-ray disks. The processor and the memory can be supplemented by and/or incorporated in special purpose logic circuitry.

To provide for interaction with a user, the above described techniques can be implemented on a computing device in communication with a display device, e.g., a CRT (cathode ray tube), plasma, or LCD (liquid crystal display) monitor, a mobile device display or screen, a holographic device and/or projector, for displaying information to the user and a keyboard and a pointing device, e.g., a mouse, a trackball, a touchpad, or a motion sensor, by which the user can provide input to the computer (e.g., interact with a user interface element). Other kinds of devices can be used to provide for interaction with a user as well; for example, feedback provided to the user can be any form of sensory feedback, e.g., visual feedback, auditory feedback, or tactile feedback; and input from the user can be received in any form, including acoustic, speech, and/or tactile input.

The above described techniques can be implemented in a distributed computing system that includes a back-end component. The back-end component can, for example, be a data server, a middleware component, and/or an application server. The above described techniques can be implemented in a distributed computing system that includes a front-end component. The front-end component can, for example, be a client computer having a graphical user interface, a Web browser through which a user can interact with an example implementation, and/or other graphical user interfaces for a transmitting device. The above described techniques can be implemented in a distributed computing system that includes any combination of such back-end, middleware, or front-end components.

The components of the computing system can be interconnected by transmission medium, which can include any form or medium of digital or analog data communication (e.g., a communication network). Transmission medium can include one or more packet-based networks and/or one or more circuit-based networks in any configuration. Packet-based networks can include, for example, the Internet, a carrier internet protocol (IP) network (e.g., local area network (LAN), wide area network (WAN), campus area network (CAN), metropolitan area network (MAN), home area network (HAN)), a private IP network, an IP private branch exchange (IPBX), a wireless network (e.g., radio access network (RAN), Bluetooth, Wi-Fi, WiMAX, general packet radio service (GPRS) network, HiperLAN), and/or other packet-based networks. Circuit-based networks can include, for example, the public switched telephone network (PSTN), a legacy private branch exchange (PBX), a wireless network (e.g., RAN, code-division multiple access (CDMA) network, time division multiple access (TDMA) network, global system for mobile communications (GSM) network), and/or other circuit-based networks.

Information transfer over transmission medium can be based on one or more communication protocols. Communication protocols can include, for example, Ethernet protocol, Internet Protocol (IP), Voice over IP (VOIP), a Peer-to-Peer (P2P) protocol, Hypertext Transfer Protocol (HTTP), Session Initiation Protocol (SIP), H.323, Media Gateway Control Protocol (MGCP), Signaling System #7 (SS7), a Global System for Mobile Communications (GSM) protocol, a Push-to-Talk (PTT) protocol, a PTT over Cellular (POC) protocol, Universal Mobile Telecommunications System (UMTS), 3GPP Long Term Evolution (LTE) and/or other communication protocols.

Devices of the computing system can include, for example, a computer, a computer with a browser device, a telephone, an IP phone, a mobile device (e.g., cellular phone, personal digital assistant (PDA) device, smart phone, tablet, laptop computer, electronic mail device), and/or other communication devices. The browser device includes, for example, a computer (e.g., desktop computer and/or laptop computer) with a World Wide Web browser (e.g., Chrome™ from Google, Inc., Microsoft® Internet Explorer® available from Microsoft Corporation, and/or Mozilla® Firefox available from Mozilla Corporation). Mobile computing device include, for example, a Blackberry® from Research in Motion, an iPhone® from Apple Corporation, and/or an Android™-based device. IP phones include, for example, a Cisco® Unified IP Phone 7985G and/or a Cisco® Unified Wireless Phone 7920 available from Cisco Systems, Inc.

Comprise, include, and/or plural forms of each are open ended and include the listed parts and can include additional parts that are not listed. And/or is open ended and includes one or more of the listed parts and combinations of the listed parts.

One skilled in the art will realize the invention may be embodied in other specific forms without departing from the spirit or essential characteristics thereof. The foregoing embodiments are therefore to be considered in all respects illustrative rather than limiting of the invention described herein.

What is claimed is:

1. A computerized method for authenticating an end point device participating in a media-based conference call, the method comprising:

receiving, by a call processing module of a server computing device, a request to join a conference call between a plurality of end point devices, the request including a media stream associated with the first end point device;

analyzing, by the call processing module, the media stream using a matrix of authentication factors, wherein the authentication factors include

a) user-specific factors comprising facial recognition attributes, audio recognition attributes, and user gesture attributes;

b) environment factors comprising (i) acoustic attributes of a device used to capture audio during the call and of a room in which the first end point device is located, wherein the call processing module analyzes a waveform associated with the captured audio to determine the acoustic attributes, and (ii) lighting hues of the room in which the first end point device is located, wherein the call processing module analyzes video images in the media stream to determine the lighting hues; and

c) technical factors comprising technical attributes of the end point device and technical attributes of the media stream;

determining, by the call processing module, an authentication score for the first end point device based upon the media stream analysis; and

determining, by the call processing module, whether to connect the first end point device to the conference call, another media resource, or another user of a network or communication system based upon the authentication score.

2. The method of claim 1, further comprising:

determining, by the call processing module, an identity of a user of the first end point device based upon the media stream analysis; and

retrieving, by the call processing module, a user profile based upon the identity of the user, the user profile including a set of permissions associated with conference call functionality.

3. The method of claim 1, wherein the technical attributes of the first end point device include an image resolution of a camera coupled to the first end point device, a device identifier, a location, and an originating address.

4. The method of claim 1, wherein the technical attributes of the media stream include a media format and a media quality.

5. The method of claim 1, further comprising:

transmitting, by the call processing module, the media stream associated with the first end point device to one or more other end point devices connected to the conference call;

receiving, by the call processing module, a validation signal from the one or more other end point devices to confirm an identity of a user of the first end point device; and

adding, by the call processing module, the received validation signal to the matrix of authentication factors.

6. The method of claim 1, wherein the media stream analysis includes comparing the matrix of authentication factors for the media stream to a matrix of authentication factors for prior media streams associated with the first end point device.

7. The method of claim 6, wherein the media stream analysis includes analyzing the matrix of authentication factors for the media stream based upon user- and/or system-specified preferences.

8. The method of claim 1, further comprising storing, by the call processing module, the media stream analysis and authentication score for the media stream in a database.

9. The method of claim 1, wherein the media stream analysis and authentication score determination occur periodically during the conference call.

10. The method of claim 1, wherein the authentication score is below a predetermined threshold, the method further comprising transmitting, by the call processing module, a request for user credentials to the first end point device.

11. The method of claim 1, further comprising transmitting, by the call processing module, the authentication score for the first end point device for display on one or more other end point devices connected to the conference call.

12. The method of claim 1, further comprising:

denying, by the call processing module, the request to join the conference call when the authentication score is below a predetermined threshold; and

disconnecting, by the call processing module, the first end point device.

13. The method of claim 1, further comprising:

allowing, by the call processing module, the request to join the conference call when the authentication score is at or above a predetermined threshold; and

connecting, by the call processing module, the first end point device to the conference call.

14. A system for authenticating an end point device participating in a media-based conference call, the system comprising a server computing device having a call processing module configured to:

receive a request to join a conference call between a plurality of end point devices, the request including a media stream associated with the first end point device;

analyze the media stream using a matrix of authentication factors, wherein the authentication factors include

a) user-specific factors comprising facial recognition attributes, audio recognition attributes, and user gesture attributes;

b) environment factors comprising (i) acoustic attributes of a device used to capture audio during the call and of a room in which the first end point device is located, wherein the call processing module analyzes a waveform associated with the captured audio to determine the acoustic attributes, and (ii) lighting hues of the room in which the first end point device is located, wherein the call processing module analyzes video images in the media stream to determine the lighting hues; and

c) technical factors comprising technical attributes of the end point device and technical attributes of the media stream;

determine an authentication score for the first end point device based upon the media stream analysis; and

determine whether to connect the first end point device to the conference call, another media resource, or another user of a network or communication system based upon the authentication score.

15. The system of claim 14, wherein the call processing module is further configured to:

determine an identity of a user of the first end point device based upon the media stream analysis; and

retrieve a user profile based upon the identity of the user, the user profile including a set of permissions associated with conference call functionality.

16. The system of claim 14, wherein the technical attributes of the first end point device include an image resolution of a camera coupled to the first end point device, a device identifier, a location, and an originating address.

17. The system of claim 14, wherein the technical attributes of the media stream include a media format and a media quality.

18. The system of claim 14, wherein the call processing module is further configured to:

transmit the media stream associated with the first end point device to one or more other end point devices connected to the conference call;

receive a validation signal from the one or more other end point devices to confirm an identity of a user of the first end point device; and

add the received validation signal to the matrix of authentication factors.

19. The system of claim 14, wherein the media stream analysis includes comparing the matrix of authentication factors for the media stream to a matrix of authentication factors for prior media streams associated with the first end point device.

20. The method of claim 19, wherein the media stream analysis includes analyzing the matrix of authentication factors for the media stream based upon user- and/or system-specified preferences.

21. The system of claim 14, wherein the call processing module is further configured to store the media stream analysis and authentication score for the media stream in a database.

22. The system of claim 14, wherein the media stream analysis and authentication score determination occur periodically during the conference call.

23. The system of claim 14, wherein the authentication score is below a predetermined threshold, the call processing

module further configured to transmit a request for user credentials to the first end point device.

24. The system of claim **14**, wherein the call processing module is further configured to:

deny the request to join the conference call when the authentication score is below a predetermined threshold; and

disconnect the first end point device.

25. The system of claim **14**, wherein the call processing module is further configured to:

allow the request to join the conference call when the authentication score is at or above a predetermined threshold; and

connect the first end point device to the conference call.

26. A computer program product, tangibly embodied in a non-transitory computer readable storage device, for authenticating an end point device participating in a media-based conference call, the computer program product including instructions operable to cause a call processing module of a server computing device to:

receive a request to join a conference call between a plurality of end point devices, the request including a media stream associated with the first end point device;

analyze the media stream using a matrix of authentication factors, wherein the authentication factors include

a) user-specific factors comprising facial recognition attributes, audio recognition attributes, and user gesture attributes;

b) environment factors comprising (i) acoustic attributes of a device used to capture audio during the call and of a room in which the first end point device is located, wherein the call processing module analyzes a waveform associated with the captured audio to determine the acoustic attributes, and (ii) lighting hues of the room in which the first end point device is located, wherein the call processing module analyzes video images in the media stream to determine the lighting hues; and

c) technical factors comprising technical attributes of the end point device and technical attributes of the media stream;

determine an authentication score for the first end point device based upon the media stream analysis; and

determine whether to connect the first end point device to the conference call, another media resource, or another user of a network or communication system based upon the authentication score.

* * * * *